ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/340939215

A Review of an Old Dilemma: Demosaicking First, or Denoising First?

Preprint · April 2020

CITATIONS
0

READS
160

3 authors:

Qiyu Jin
Inner Mongolia University
31 PUBLICATIONS **104** CITATIONS
SEE PROFILE

Gabriele Facciolo
École normale supérieure Paris-Saclay
107 PUBLICATIONS **1,574** CITATIONS
SEE PROFILE

Jean-Michel Morel
Ecole normale supérieure de Cachan
468 PUBLICATIONS **30,622** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Image and video inpainting View project

Project PSF Estimation View project

All content following this page was uploaded by Gabriele Facciolo on 04 August 2020.

The user has requested enhancement of the downloaded file.

arXiv:2004.11577v1 [eess.IV] 24 Apr 2020

A Review of an Old Dilemma: Demosaicking First, or Denoising First?

Qiyu Jin
School of mathematical science,
Inner Mongolia University
qyjin2015@aliyun.com

Gabriele Facciolo
Centre Borelli,
ENS Paris-Saclay, CNRS
facciolo@cmla.ens-cachan.fr

Jean-Michel Morel
Centre Borelli,
ENS Paris-Saclay, CNRS
moreljeanmichel@gmail.com

April 27, 2020

Abstract

Image denoising and demosaicking are the most important early stages in digital camera pipelines. They constitute a severely ill-posed problem that aims at reconstructing a full color image from a noisy color filter array (CFA) image. In most of the literature, denoising and demosaicking are treated as two independent problems, without considering their interaction, or asking which should be applied first. Several recent works have started addressing them jointly in works that involve heavy weight CNNs, thus incompatible with low power portable imaging devices. Hence, the question of how to combine denoising and demosaicking to reconstruct full color images remains very relevant: Is denoising to be applied first, or should that be demosaicking first? In this paper, we review the main variants of these strategies and carry-out an extensive evaluation to find the best way to reconstruct full color images from a noisy mosaic. We conclude that demosaicking should applied first, followed by denoising. Yet we prove that this requires an adaptation of classic denoising algorithms to demosaicked noise, which we justify and specify.

1 Introduction

Most digital cameras capture image data by using a single sensor coupled with a color filter array (CFA). At each pixel in the array, only one color component is recorded and the resulting image is called a mosaic. The most common CFA is the Bayer color array [6], in which two out of four pixels measure the green component, one measures the red and one the blue. The process of completing the missing red, green and blue values at each pixel is called demosaicking. Noise is inevitable, especially in low light conditions and for small camera sensors like those used in mobile phones. The conventional approach in image restoration pipelines for processing noisy raw sensor data has long been to apply denoising and demosaicking as two independent steps [46]. Furthermore, the immense majority of image processing papers addressing one of both operations do not address its combination with the other one. All classic denoising algorithms have been designed for color or grey level images with white noise added. Yet the realistic data are different: either a mosaic with white noise or a demosaicked image with structured noise.

Joint denoising/demosaicking methods. This has led several recent works to propose joint demosaicking- de-

noising methods [9, 20, 26, 32]. For example [21] proposed a variational model to solve jointly demosaicking, denoising and deblurring. It uses a sparsifying prior based on wavelet packets and applied on decorrelated color channels. More detail about the technicalities of this sophisticated method can be found in [2]. Life has become far easier for joint denoising/demosaicking with the emergence of machine learning methods. It is, indeed, easy to simulate as many learning data as needed. This methodology can be used to obtain ground breaking demosaicking algorithms such as [52]. This paper proposed in 2018 a demosaicking CNN outperforming the best handcrafted algorithms including ARI [45] by nearly 2 decibels. In [32] a public ground truth dataset was introduced and used for one of the first joint demosaicking and denoising methods based on machine learning. In a rapid succession, two state of the art denoising+demosaicking methods based on deep learning were proposed: [20] and [37]. This last paper performs joint denoising and demosaicking by a customized neural network presented as a cascade of energy minimization methods tuned by learning. The outstanding results beat the claimed anterior best method [25] by 1 decibel. Then in 2018 we have two still better performing methods, [15] involving a GAN, which compares favorably to [20] and [54]. The method recently proposed in [38] performs joint denoising and demosaicking by inserting many residual denoising layers in a CNN. This complex method is claimed to beat [20] and [37] by a good margin. Lastly, in 2019, [17] introduced a "mosaic-to mosaic" training strategy analog to the noise-to-noise [41] and frame-to-frame [18] frameworks to handle noisy mosaicked raw data, and training both demosaicking and joint denoising and demosaicking networks without requiring ground truth. The method starts from pairs or bursts of raw images of the same scene. It registers them and learns to predict the missing colors.

Yet the question of how to combine denoising and demosaicking algorithms conceived as independent blocks remains very relevant, especially in the context of low power or portable devices, and given the fact that the main effort in denoising and demosaicking has addressed them independently. A big argument in favour of performing denoising before demosaicking is that most existing demosaicking algorithms have been developed under the unrealistic assumption of noise-free data [8, 10, 20, 23, 24, 28, 33–36, 38, 42, 45, 49, 53–56, 60]. Yet the performance of these algorithms can degrade dramatically when the noise level increases on the CFA raw image. Therefore, a previous denoising step is implicitly required by these algorithms.

In this paper we focus on the early CFA processing in the imaging pipeline (operating in linear space). We assume that the noise in the raw mosaic is additive white Gaussian (AWGN) and that its variance is known. This is realistic because, first, a *variance stabilizing transform* (VST) [5] applied to a raw image results in a nearly AWG noise and, second, because an accurate noise model is often known or can be estimated [11, 50]. In general, image denoising methods can be grouped into two major categories, the model based methods such as non-local means [7, 29, 30], nlBayes [39], CBM3D [12] and WNNM [22], and deep learning methods such as [27, 57]. The ensuing CNNs are sometimes flexible in handling denoising problems with various noise levels.

Our goal here is to determine which strategy is more advantageous for coupling demosaicking and denoising : Is it applying denoising and then demosaicking (which we will denote $DN\&DM$: DN and DM indicate denoising and demosaicking respectively), or is it better to apply first demosaicking and then denoising ($DM\&DN$)?

DN&DM methods (*i.e.* denoising then demosaicking): advantages and drawbacks. Many state of the art works [31, 46, 47, 58] support the opinion that $DN\&DM$ outperforms $DM\&DN$. Their first convincing argument is that after demosaicking noise becomes correlated, thus losing its independent identically distributed (i.i.d.) white Gaussian property. This increases the difficulty of applying efficient denoising and actually seems to discard all classic algorithms, that mostly rely on the AGWN assumption. A second obvious argument is that the best demosaicking algorithms have been designed with noise-free images.

For example, Park *et al.* [47] considered the classic Hamilton-Adams (HA) [23] and [16] for demosaicking, combined with two denoising methods, BLS-GSM [51] and CBM3D [13]. This combination raises the question of adapting CBM3D to a CFA. To do so, the authors apply a sparsifying 4D color transform to the 4-channel image formed by rearranging the Bayer pixels, apply BM3D to each channel, then apply the inverse color transform.

In the very same vein, in the BM3D-CFA method [14] BM3D is applied directly on the CFA color array. To do so, "only blocks having the same CFA configuration are being compared to build the 3D blocks. This is the only modification of the original BM3D". A little thought leads to the conclusion that this amounts to denoise four different mosaics of the same image before aggregating the four values obtained for each pixel. The authors compare two denoising algorithms with two different setups: a) filtering CFA as a single image and b) splitting the CFA into four color components, filtering them separately, and recombining back the denoised CFA image. This paper showed a systematic improvement over [58]. They use Zhang-Wu [59] as demosaicking method for their comparison of result after demosaicking. In our comparisons the method of [14] will be mentioned every time we consider the $DN\&DM$ setup with BM3D. We will nevertheless replace the demosaicking of [59] by RCNN [54] or RI [33], which clearly outperform it.

Similarly in [9] denoising is performed by an adaptation of NL-means to the Bayer pattern, where only patches with the same CFA configuration are being matched. This paper formulates the demosaicking as a super-resolution problem, assuming that the observed values are actually averages of four values in the high resolution image. It then guides this super-resolution problem by the NL-means weights. The method is compared with [44] and [58]. The authors of [58] also propose an $DN\&DM$ method, where the demosaicking method is [59] and the the denoising method is an adaptation of nlBayes [39] to a Bayer pattern. First, the method extracts blocks with similar configuration in the Bayer array and groups them by similarity, then it applies to them PCA and a Wiener denoising procedure which can be also interpreted as an LMMSE. In our experiments, this PCA method [58] will be considered every time we evaluate the $DN\&DM$ scheme (but combined with a more recent demosaicking algorithm such as RCNN [54]). The more recent paper [61] involves similar arguments. This paper uses [4], a linear filter to extract the luminance from the CFA. Then it remarks that this luminance is correlated, so it applies a variant of NL-means that attempts to decorrelate the noise. The same method is applied to each downsampled color channel and the high frequency of the grey level is transported back to the color channels. This method under-performs with respect to others considered here, so we shall not include it to our final comparison tables. Nevertheless, it remains of interest as a fast method compatible with low power cameras. The paper proves that it has a performance very close to a combination of [58] and [26].

The paper [48] is another method promoting denoising before demosaicking, involving dictionary learning methods to remove the Poisson noise from the single channel images prior to demosaicing. Experimental results on simulated noisy images as well as real camera acquisitions, show the advantage of these methods over approaches that remove noise subsequent to demosaicing. The paper nevertheless uses [43] which is a historic but outdated demosaicking method.

To summarize, in the $DN\&DM$ strategy all classic denoising algorithms such as CBM3D, nlBayes, nlMeans have been adapted to handle a noisy mosaic where only one of R, G or B is known at each pixel. Several of them [31, 46, 47, 58] address this realistic case by processing the noisy CFA images as a half-size 4-channel color image (with one red, two green and one blue channels) and then apply a multichannel denoising algorithm to it. The advantage of the denoising step of $DN\&DM$ is that the Poisson noise can be led back by the classic Anscombe transform to the case of i.i.d. white Gaussian, and the disadvantage is that the resolution of the image is reduced and, as a result, some details might be lost after denoising. Another issue of this strategy is that the spatial relative positions of the R, G, and B pixels are lost by handling the image as a four channel half size image.

In this paper, we address the above mentioned issues. We shall first delve into the advantages and disadvantages of $DN\&DM$ and $DM\&DN$ approaches. We shall then analyze noise properties after demosaicking and adjust two existing classic denoising algorithms (CBM3D and nlBayes) to accommodate them to this type of noise. Then, we shall perform a thorough experimental evaluation that will lead us to conclude that $DM\&DN$ (with an adjusted noise parameter) is superior to $DN\&DM$. This result is opposite to the conclusion of [31, 46, 47, 58]. The advantages of $DM\&DN$ seem to be linked to the fact that this scheme does not handle a half size 4-channels color image; it therefore uses the classic denoising methods directly on a full resolution color image; this results in more details being preserved and avoids checkerboard effects.

The rest of the paper is structured as follows. In Sec-

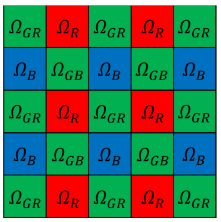

Ω_{GR} Ω_R Ω_{GR} Ω_R Ω_{GR}
Ω_B Ω_{GB} Ω_B Ω_{GB} Ω_B
Ω_{GR} Ω_R Ω_{GR} Ω_R Ω_{GR}
Ω_B Ω_{GB} Ω_B Ω_{GB} Ω_B
Ω_{GR} Ω_R Ω_{GR} Ω_R Ω_{GR}

Figure 1: Bayer color filter array, CFA, which is used by most cameras.

tion 2 we present in detail the problem and the main ideas behind the proposed demosaicking and denoising strategy. Section 3 is a detailed evaluation of the proposed strategy. Section 4 concludes.

2 The demosaicking and denoising framework

In a single-sensor camera equipped with a color filter array (CFA) [6], only one pixel value among the three RGB values is recorded at each pixel. Consider a CFA block as shown in Fig. 1. The raw Bayer CFA images are scalar mosaic matrices with noise. Obtaining high quality color images requires completing the missing color channels and removing the noise. As mentioned in the introduction, for this task we will consider two main schemes: $DM\&DN$ (demosaicking then denoising) and $DN\&DM$ (denoising then demosaicking).

Park *et al.* [47] argued that demosaicking introduces chromatic and spatial correlations to the noisy CFA image. Then the noise is no longer i.i.d. white Gaussian, which makes it harder to remove. In [31], some experiments were done to show that $DN\&DM$ schemes are more efficient to suppress noise than $DM\&DN$ schemes. Based on this argument several denoising methods [3, 40, 47, 58] for raw CFA images before demosaicking were introduced. Other denoising methods that are not explicitly designed to handle raw CFA images (such as CBM3D and nlBayes) can also be applied on noisy CFA images by rearranging the CFA image into a half-size four-channels image with two greens on which the denoising algorithm is applied [47]. The denoised CFA is then recovered by undoing the pixel rearrangement. However, this strategy reduces the resolution of the image seen by the denoiser, and we observed checkerboard effects resulting from chromatic aberrations in the two green channels after denoising. To address this issue, Danielyan *et al.* [14] proposed BM3D-CFA which amounts to denoise four different mosaics of the same image before aggregating the four values obtained for each pixel.

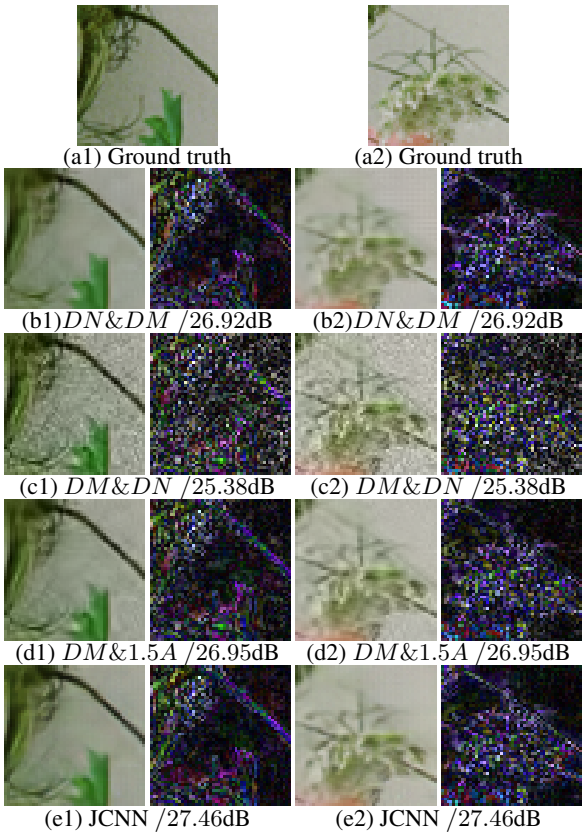


Figure 2: Comparison of different denoising and demosaicking schemes with noise $\sigma_0 = 20$. In each experiment: left, detail of the demosaicked and denoised image; right, the difference with original that should contain mainly noise. DN: CBM3D denoising; DM: demosaicking (here we use RCNN). $1.5DN$ means that if noise level is σ_0, the input noise level parameter of denoising method DN is $\sigma = 1.5\sigma_0$; $DN\&DM$: uses the BM3D-CFA framework [14] for denoising.

Modeling demosaicking noise. In order to solve the above two problems, we shall revisit the $DM\&DN$ scheme. Compared to the $DN\&DM$ scheme, the advantage of $DM\&DN$ is that it does not halve the image size. This is a way around the above mentioned problems. A serious drawback, though, is that chromatic and spatial correlations have been introduced by the demosaicking in the noise of the CFA image. The result is that the noise is no longer white. We next analyze some properties of the *demosaicked noise*.

Definition *Given a ground truth color image (R, G, B) we define the* demosaicked noise *associated with a demosaicking method DM in the following way: first the image is mosaicked so that only one value of either R, G, DM is kept at each pixel, according to a fixed Bayer pattern. Then white noise with standard deviation σ_0 is added to the mosaicked image, and the resulting noisy mosaic is demosaicked by DM, hence giving a noisy image $(\tilde{R}, \tilde{G}, \tilde{B})$. We call demosaicked noise the difference $(\tilde{R} - R, \tilde{G} - G, \tilde{B} - B)$. In short, it is the difference between the demosaicked version of a noisy image and its underlying ground truth.*

The model of the demosaicked noise depends on the choice of the demosaicking algorithm DM. For the demosaicking step we will evaluate the following state of the art methods, which have an increasing complexity: HA [23], RI [33], MLRI [34], ARI [45], LSSC [42], RCNN [54] and JCNN [20]. We are interested in algorithms with low or moderate power; only HA, RI, MLRI and RCNN have a reasonable complexity in this context. For the denoising step we shall likewise consider two classic hand-crafted algorithms, CBM3D and nlBayes.

Fig. 2 (c1) and (c2) shows an example where noisy CFA images with noise of standard deviation σ_0 were first demosaicked by RCNN and then restored by CBM3D assuming a noise parameter $\sigma = \sigma_0$. The output of CBM3D with $\sigma = \sigma_0$ has a strong residual noise. Similar results are also obtained with nlBayes (see the supplementary material). To understand empirically the right noise model to adopt after demosaicking, we simulated this $DM\&DN$ pipeline for different levels of noise σ_0, and applied CBM3D after demosaicking with a noise parameter corresponding to σ_0 multiplied by different factors $(1.0, 1.1, \cdots, 1.9)$.

Table 1: Denoising after demosaicking $DM\&DN$, where DN is CBM3D [13] with noise parameter equal to $C\,\sigma_0$, while noise in the raw image has standard deviation $\sigma_0 = 20$. Each row shows the CPSNR result for C ranging from 1.0 to 1.9. Each column corresponds to a different demosaicking method DM. The best result of each column is marked with a box. The best result of each line is in **red** and the second best one is in **green**. The best factor C for all methods is $C \simeq 1.5$, the same is true for different values of σ_0 as well (see supplementary material).

C	HA	GBTF	RI	MLRI	ARI	LSSC	RCNN
1.0	28.15	27.58	28.46	27.95	28.70	27.19	27.28
1.1	28.56	28.15	28.83	28.44	28.98	27.89	28.05
1.2	28.85	28.55	29.08	28.80	29.18	28.43	28.67
1.3	29.05	28.81	29.23	29.03	29.29	28.78	29.09
1.4	29.18	28.96	**29.31**	29.17	**29.35**	29.00	29.34
1.5	**29.23**	**29.00**	**29.32**	**29.22**	**29.35**	**29.06**	**29.41**
1.6	**29.25**	**29.01**	**29.30**	**29.23**	**29.33**	**29.06**	**29.41**
1.7	**29.25**	**28.97**	29.26	**29.20**	29.29	**29.02**	**29.36**
1.8	29.22	28.92	29.20	29.15	29.23	28.95	29.28
1.9	29.17	28.85	29.13	29.08	29.17	28.88	29.20

Table 2: RMSE between original and demosaicked image for different demosaicking algorithms in presence of noise of standard deviation σ_0.

σ_0	HA	GBTF	RI	MLRI	ARI	LSSC	RCNN
1	5.04	5.10	4.17	4.06	3.72	4.40	3.21
5	6.78	6.87	6.12	6.10	5.74	6.36	5.59
10	10.18	10.27	9.53	9.74	9.09	9.96	9.65
20	17.75	17.83	16.77	17.56	16.06	18.16	18.04
40	32.67	32.76	30.77	32.64	29.36	33.68	33.98
60	46.14	46.35	43.43	46.11	41.44	48.11	47.95

The results are shown in Table 1, where the classic color peak signal-to-noise ratio (CPSNR) [4] is adopted as a logarithmic measure of the performance of the algorithms. It is defined by

$$\mathrm{CPSNR} = 10 \log_{10} \frac{255^2}{\sum_{X=R,G,B} \mathrm{MSE}(X)/3}, \quad \text{with}$$
$$\mathrm{MSE}(X) = \frac{1}{|\Omega|} \sum_{(i,j)\in\Omega} (\widehat{X}(i,j) - X(i,j))^2,$$

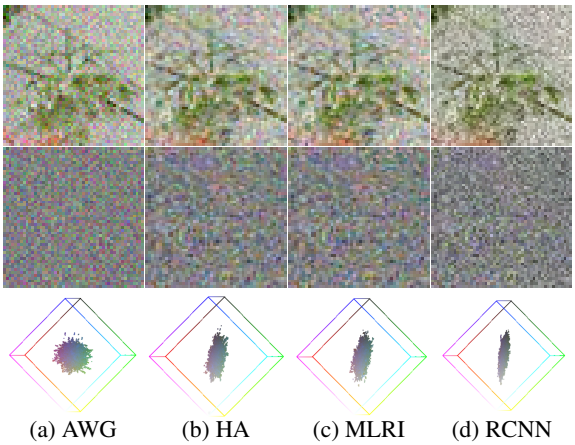
where X denotes the ground truth image and $\widehat{X}$ is the estimated color image. From 1.0 to 1.9, the CPSNR increases first and then decreases. We can see that the best values are distributed on the lines with factors from 1.4 to 1.7. A similar behavior was also observed using nlBayes for denoising as well as for other levels of noise (see the supplementary material).

This does not mean that the overall noise standard deviation has increased after demosaicking. Let us consider the noise standard deviation estimated as the mean RMSE of the demosaicked images from the Imax [60] dataset with different noise levels, given in Table 2. We observe that for low noise ($\sigma_0 = 1$) there is a serious demosaicking error, of about 4, not caused by the noise, but by the demosaicking itself. However, for $\sigma_0 > 10$ we see that the RMSE of the demosaicked image tends to roughly 3/4 of the initial noise standard deviation.

At first sight, this $3/4$ factor seems to contradict the observation that denoising with a parameter $1.5\sigma_0$ yields better results. This leads us to analyzing the structure of the residual noise. Fig. 3 shows an image contaminated with AWG noise with standard deviation $\sigma_0 = 20$ and its resulting demosaicked noise for respectively HA, MLRI, RCNN. In the last row of the figure, one can observe the color spaces (in standard (R,G,B) Cartesian coordinates) of each of these noises, each cloud being presented in its projection with maximal area. As expected, the AWG color space is isotropic and has an apparent diameter proportional to $4\sigma_0 \simeq 80$. The color space of the demosaicked noise is instead elongated in the luminance direction $Y = \frac{R+G+B}{\sqrt{3}}$ to about $6\sigma_0 \simeq 120$ and squeezed in the others. This amounts to an increased noise standard deviation for Y after demosaicking, and much less noise in the chromatic directions.

This is confirmed by Table 3 that shows variances and covariances of (R, G, B) and (Y, U, V) respectively for an AWG noise with $\sigma_0 = 20$, and then for the demosaicked noise obtained from it after demosaiciking with RI, MLRI and RCNN. In Table 3 (a) these statistics are computed on a pure white noise image with $\sigma = 20$. Hence the variance of Y is 400, as the $(R, G, B) \to (Y, U, V)$ transform is implemented as an isometry of $\mathbf{R}^3$. The variance of Y is a growing sequence for the demosaicked noise obtained by increasingly sophisticated demosaicking: 715 for RI, 772 for MLRI, 972 for RCNN. In contrast, the demosaicked noise is reduced in the U and V axes, with its variance passing from 400 for AWGN to 168 and 94 for RI, and even down to 43 and 55 for RCNN. Hence, the noise standard deviation on U or V has been divided by a factor between 2 and 3. But Table 3 also shows that the residual noise on U and V is strongly spatially correlated, it is therefore a low frequency noise, that will require stronger filtering than white noise to be removed. This table also shows that the Y component of the demosaicked noise remains almost white.

This leads to a simple conclusion: since image denois-



(a) AWG (b) HA (c) MLRI (d) RCNN

Figure 3: AWG noise image and demosaicking noise with standard deviation $\sigma = 20$ for respectively HA, MLRI, RCNN. Last row: the color spaces (in standard (R,G,B) Cartesian coordinates) of each noise, presented in their projection with maximal area. As expected, the AWG color space is isotropic, while the color space after demosaicking is elongated in the luminance direction Y and squeezed in the others. This amounts to an increased noise standard deviation for Y after demosaicking, and less noise in the chromatic directions.

ing algorithms are guided by the Y component [13, 39], we can denoise with methods designed for white noise, but with a noise parameter adapted to the increased variance of Y.

To understand why the variance of Y is far larger than the AWG noise it comes from, let us study in Table 4 the correlation between the three channels (R, G, B) in the demosaicked noise of RI, MLRI and RCNN. We observe a strong (R, G, B) correlation ranging from 0.6 for RI to 0.89 for RCNN, which is caused by the "tendency to grey" of all demosaicking algorithms. Assuming that the demosaicked noisy pixel components (denoted $\widetilde{\epsilon}_R, \widetilde{\epsilon}_G, \widetilde{\epsilon}_B$) have a correlation coefficient close to 1 then we have

$$Y = \frac{\widetilde{\epsilon}_R + \widetilde{\epsilon}_G + \widetilde{\epsilon}_B}{\sqrt{3}} \sim \sqrt{3}\, N(0, \sigma_0).$$

This factor of about 1.7 corresponds to the case with maximum correlation. Our empirical observation of an optimal factor near 1.5 responds to a lower correlation between the colors.

3 Experimental evaluation

To evaluate the proposed framework for denoising and demosaicking, we shall use two classic noise free color image datasets: Kodak and Imax. The Imax dataset [60] consists of 18 images of 500×500 pixels, cropped from high-resolution images of size: 2310×1814. The Kodak dataset consists of 25 images of 768×512 pixels released by the Kodak Corporation for unrestricted research usage[1]. We also evaluated it on a set of 14 real raw images from the SIDD dataset [1], which comes with ground truth acquisitions.

Evaluation of DN&DM and DM&DN strategies. We performed simulations with the schemes: $DN\&DM$ and $DM\&DN$. The considered demosaicking methods range from classic to very modern: HA[23], RI[33], MLRI[34], ARI [45], and RCNN[54]. For the denoising stage two classic hand-crafted patch-based denoising algorithms were considered: CBM3D [13] and nlBayes [39]. As commented in the introduction, both methods can be adapted to handle mosaics (in the $DN\&DM$ setting). In

[1]Image source: http://r0k.us/graphics/kodak

Table 3: Variance and covariance of (R, G, B) and (Y, U, V) (each first row) and the corresponding correlations (each second row) between pixels (i, j) and $(i + s, j + t)$, $s, t = 0, 1, 2$ first for AWGN (a) with standard deviation $\sigma = 20$, then for its demosaicked versions by RI (b), MLRI (c) and RCNN (d).

	(*i*,*j*)	(*i*,*j*+1)	(*i*,*j*+2)	(*i*+1,*j*)	(*i*+1,*j*+1)	(*i*+1,*j*+2)	(*i*+2,*j*)	(*i*+2,*j*+1)	(*i*+2,*j*+2)
R	400.6	0.6	0.4	0.7	0.1	0.7	0.3	0.2	0.8
G	401.7	0.5	1.1	0.1	0.3	0.9	1.0	0.6	0.4
B	400.2	1.2	0.1	0.5	0.6	0.0	1.9	0.3	1.9
Y	399.6	1.1	0.1	0.3	0.1	0.9	0.2	0.5	1.2
U	401.5	0.1	0.8	0.6	0.3	0.3	0.9	0.5	1.3
V	401.4	0.2	1.8	0.9	0.2	1.0	0.6	0.2	0.2

(a) AWG noise

	(*i*,*j*)	(*i*,*j*+1)	(*i*,*j*+2)	(*i*+1,*j*)	(*i*+1,*j*+1)	(*i*+1,*j*+2)	(*i*+2,*j*)	(*i*+2,*j*+1)	(*i*+2,*j*+2)
R	336.4	126.8	19.4	129.9	52.9	21.6	20.7	22.4	18.7
G	295.5	92.5	0.5	95.6	20.6	1.8	0.7	1.5	4.3
B	350.5	125.9	18.1	130.4	50.7	20.8	20.0	20.9	17.5
Y	715.6	170.9	32.3	178.6	2.6	5.4	34.0	7.1	20.5
U	168.4	108.3	41.3	110.1	73.4	28.2	44.1	29.4	9.7
V	98.3	66.0	27.9	67.3	48.1	21.4	29.9	22.4	10.4

(b) RI

	(*i*,*j*)	(*i*,*j*+1)	(*i*,*j*+2)	(*i*+1,*j*)	(*i*+1,*j*+1)	(*i*+1,*j*+2)	(*i*+2,*j*)	(*i*+2,*j*+1)	(*i*+2,*j*+2)
R	361.4	128.4	18.9	130.5	46.4	20.6	21.6	21.5	19.8
G	298.9	93.0	0.5	95.1	19.1	0.9	1.0	0.5	3.8
B	370.9	127.8	19.3	130.4	46.0	20.6	21.2	20.3	19.0
Y	772.2	177.7	33.0	181.3	9.6	9.2	32.6	10.9	21.4
U	164.8	107.1	43.7	108.8	72.8	29.3	46.1	30.2	10.1
V	94.3	64.4	28.1	65.8	48.2	21.9	30.3	23.1	11.1

(c) MLRI

	(*i*,*j*)	(*i*,*j*+1)	(*i*,*j*+2)	(*i*+1,*j*)	(*i*+1,*j*+1)	(*i*+1,*j*+2)	(*i*+2,*j*)	(*i*+2,*j*+1)	(*i*+2,*j*+2)
R	359.9	47.8	5.0	51.9	21.8	17.8	5.1	19.4	9.2
G	354.8	32.6	4.4	36.3	5.8	8.4	6.4	8.8	0.6
B	356.0	49.6	6.3	53.7	23.6	18.8	7.3	19.4	9.2
Y	972.3	69.0	20.8	76.4	3.6	18.6	28.9	17.3	2.2
U	55.1	33.8	15.3	36.0	26.1	14.6	19.0	16.6	11.8
V	43.3	27.3	12.3	29.4	21.5	11.7	16.0	13.7	9.4

(d) RCNN

Table 4: Covariances (each first row) and *correlations* (each second row) of the three color channels (R, G, B) of the demosaicked noise, when the initial CFA white noise satisfies $\sigma_0 = 20$

	R	G	B
R	336.44	206.29	175.01
	1.0000	*0.6542*	*0.5097*
G	206.29	295.54	200.96
	0.6542	*1.0000*	*0.6244*
B	175.01	200.96	350.46
	0.5097	*0.6244*	*1.0000*

(a) RI

	R	G	B
R	361.42	224.39	201.41
	1.0000	*0.6826*	*0.5501*
G	224.39	298.94	216.86
	0.6826	*1.0000*	*0.6512*
B	201.41	216.86	370.92
	0.5501	*0.6512*	*1.0000*

(b) MLRI

	R	G	B
R	359.90	320.44	302.85
	1.0000	*0.8967*	*0.8461*
G	320.44	354.83	299.85
	0.8967	*1.0000*	*0.8437*
B	302.85	299.85	355.99
	0.8461	*0.8437*	*1.0000*

(c) RCNN

	R	G	B
R	334.84	297.31	275.28
	1.0000	*0.8675*	*0.8181*
G	297.31	350.81	270.32
	0.8675	*1.0000*	*0.7848*
B	275.28	270.32	338.17
	0.8181	*0.7848*	*1.0000*

(d) JCNN

the case of CBM3D this amounts to applying the method by Danielyan *et al.* [14], while for nlBayes this is done by denoising the 4-channel image associated to the mosaic.

The denoising and demosaicking schemes with the above mentioned demosaicking algorithms and denoising methods were applied to the mosaic images of the Imax image dataset corrupted by additive white Gaussian noise with standard deviations $\sigma_0 = 1, 3, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60$.

Due to space constraints, in Table 5 we only report the results corresponding to one noise level $\sigma_0 = 20$. Results corresponding to other noise levels are in the supplementary material. From Table 5, we can see that $DM\&DN$ with parameter $\sigma = \sigma_0$ is not better than $DN\&DM$, but $DM\&1.5DN$ (which denotes denoising DN with parameter $\sigma = 1.5\sigma_0$) beats clearly $DN\&DM$.This might explain why many researchers think that the scheme $DN\&DM$ was superior to the scheme $DM\&DN$.

Table 5: Comparison in CPSNR(dB) of average restoration performance between $DN\&DM$ and $DM\&DN$ for a fixed level of noise $\sigma_0 = 20$. We test two denoisers DN namely CBM3D, and nlBayes, and $1.5DN$ means that if noise level is σ_0, the noise level parameter for the denoising method DN is $\sigma = 1.5\sigma_0$. Both denoisers can be adapted to handle mosaics in the $DN\&DM$ schemes (see in the text). The best result of each column is marked with a box. The best result of each line is in **red** and the second best one is in **green**.

DN	Algorithm	HA	RI	MLRI	ARI	RCNN
CBM3D	$DN\&DM$	28.11	28.45	27.97	**28.69**	27.27
	$DM\&DN$	28.15	**28.46**	27.95	**28.70**	27.28
	$DM\&1.5DN$	29.24	29.32	29.22	29.36	**29.41**
nlBayes	$DN\&DM$	28.17	28.17	28.17	28.18	**28.28**
	$DM\&DN$	28.67	**28.99**	28.57	**29.21**	28.02
	$DM\&1.5DN$	29.29	29.26	29.22	29.31	**29.36**

In addition to the good CPSNR results, one important advantage of the $DM\&DN$ schemes is the high visual quality of the final restored images. Fig. 2 demonstrates the differences between the various solutions (based on BM3D) obtained on the test image number 3 of the Imax dataset with $\sigma_0 = 20$. To save space, only crops of the full-color results and corresponding differences with the ground truth are shown here.

The $DN\&DM$ scheme shown in Fig. 2 (b1) and (b2) uses BM3D-CFA [14] for denoising; we can observe some minor checkerboard artifacts. From Fig. 2 (c1) and (c2), we can deduce that there is no checkerboard effect but that much noise remains in the restored image by $DM\&DN$ schemes with parameter $1.0\sigma_0$. The result of $DM\&1.5DN$ (Fig. 2 (d1) and (d2)) are smooth without checkerboard effects. Fig. 2 (e1) and (e2) correspond to the outputs of the CNN joint denoising and demosaicking method JCNN [20].

One can observe thin structures in the upper left corner of Fig 2 (a1), but they disappear in the restored image by $DN\&DM$. The proposed $DM\&1.5DN$ scheme restores them. The second column of Fig 2 illustrates a similar situation in which thin details are recovered by $DM\&DN$ and $DM\&1.5DN$ but not in the others.

In short, it appears that the $DM\&DN$ scheme with an appropriate parameter (namely $DM\&1.5DN$) outperforms the competition in terms of visual quality. This is due to the fact that it efficiently uses spatial and spectral image characteristics to remove noise, preserve edges and fine details. Indeed, contrary to the $DN\&DM$ schemes, $DM\&1.5DN$ does not reduce the resolution of the noisy image. Using an $DN\&DM$ scheme ends up over-smoothing the result. It comes to no surprise that JCNN performs slightly better than the other methods; however, it is much more computationally demanding and only works for $\sigma \leq 20$.

In a systematic comparison between the schemes involving CBM3D and nlBayes, schemes with CBM3D proved to perform slightly better. Furthermore, the $DM\&DN$ schemes with CBM3D are about four times faster than nlBayes. Hence, the following experiments are more focused on CBM3D.

Comparison with methods from the literature. To complete this comparison we went back to all $DN\&DM$ schemes proposed in the literature, and performed a systematic comparison for the two classic Kodak and Imax datasets. These databases are always used in demosaicking evaluations, because they illustrate different challenges of the demosaicking problem, Imax being difficult by its color contrast, and Kodak challenging for the recovery of fine structure. In Tables 6 and 7 we compare representative $DN\&DM$ methods from the literature with the best $DM\&DN$ methods identified above (all of them $DM\&1.5DN$):

– The two best performing demosaicking before denoising methods ($DM\&1.5DN$) from on Table 5 are considered. Namely, RCNN for demosaicking followed by CBM3D (denoted *RCNN+CBM3D*) or nlBayes (*RCNN+nlBayes*) for denoising.

– We also consider a "low-cost" $DM\&1.5DN$ combination using MLRI [34] for demosaicking and CBM3D for denoising (*MLRI+CBM3D*).

The considered $DN\&DM$ methods from the literature are:

– The BM3D-CFA filter was proposed in [14] to avoid the checkerboard effects resulting from independently applying BM3D to the color phases of CFA images. We evaluate BM3D-CFA [14] followed by Hamilton Adams demosaicking (denoted *BM3D+HA*), as well as followed by a state-of-the-art RCNN demosaicking [54] (*BM3D+RCNN*).

– The CFA denoising framework of Park *et al.* [47] effectively compacts the signal energy while the noise is distributed equally in all dimensions by using a color representation from the principal components analysis of the pixel RGB values in the Kodak dataset and then removes noise in each channel by BM3D. This preprocessing is advantageous for the Kodak image set, but inadequate for the Imax image set. We evaluate this framework [47] with BM3D [12] followed by the

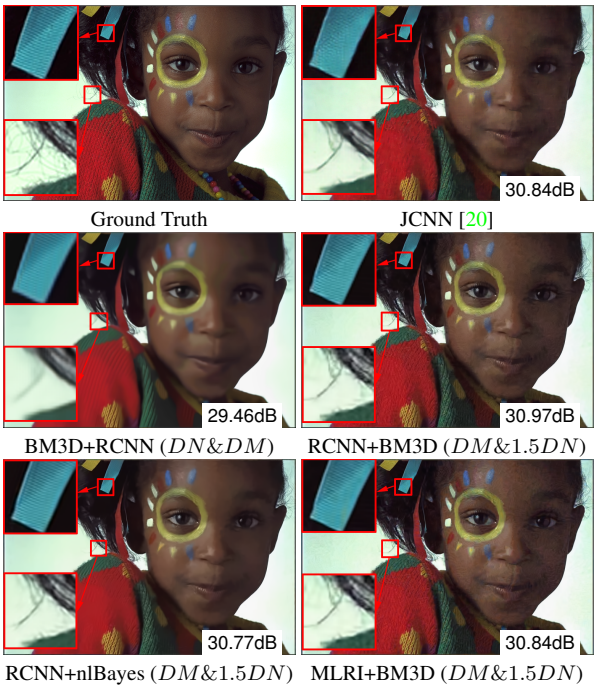


Figure 4: Demosaicking and denoising results on an image from the Kodak dataset with $\sigma = 20$. We compare an $DN\&DM$ scheme BM3D+RCNN [14], with three $DM\&1.5DN$ RCNN+CBM3D, RCNN+nlBayes and MLRI+BM3D. As a reference we also include the result of a joint CNN method JCNN [20]. But its results are only available for noise with $\sigma \leq 20$ because the network is not trained beyond that level.

Table 6: Comparison of the results (CPSNR in dB) between different denoising and demosaicking methods for the **Imax** image set. The best result of each line is in **red**, the second best one is in **green** and the third best one is in **blue**.

	$DN\&DM$					$DM\&1.5DN$			
σ	BM3D + HA	BM3D + RCNN	Park + RCNN	PCA + DLMM	PCA + RCNN	RCNN + CBM3D	RCNN + nlBayes	MLRI + CBM3D	JCNN
1	34.63	**38.53**	35.37	33.99	37.52	38.36	**38.42**	36.52	**38.59**
5	33.43	**35.62**	32.86	32.69	34.87	**35.39**	**35.29**	34.60	33.48
10	31.84	**32.92**	30.06	30.73	31.89	**32.75**	32.59	32.36	**33.09**
20	29.22	**29.55**	26.86	27.57	27.99	**29.41**	29.25	29.22	**29.79**
40	**25.50**	**25.51**	23.86	23.50	23.57	**25.52**	25.09	25.39	–
60	21.55	21.34	21.75	20.89	20.89	**22.78**	**22.31**	**22.63**	–
Av	28.09	**28.88**	26.89	26.71	27.53	**28.99**	**28.72**	28.58	–

Table 7: Comparison of the results (CPSNR in dB) between different denoising and demosaicking methods for the **Kodak** image set. The best result of each line is in **red**, the second best one is in **green** and the third best one is in **blue**.

	$DN\&DM$					$DM\&1.5DN$			
σ	BM3D + HA	BM3D + RCNN	Park + RCNN	PCA + DLMM	PCA + RCNN	RCNN + CBM3D	RCNN + nlBayes	MLRI + CBM3D	JCNN
1	34.70	40.55	40.36	38.19	39.12	**40.98**	**40.98**	38.52	**41.15**
5	32.84	34.89	34.87	34.99	35.42	**36.55**	**36.42**	**35.71**	34.13
10	30.34	30.93	30.85	31.83	32.01	**33.36**	**33.18**	32.94	**33.27**
20	27.59	27.70	27.42	28.11	28.14	**29.98**	**29.87**	29.70	**29.95**
40	24.79	24.78	24.88	24.15	24.08	**26.71**	**26.29**	**26.44**	–
60	22.58	22.55	23.19	21.77	21.70	**24.42**	**23.93**	**24.16**	–
Av	27.47	28.35	28.36	27.96	28.09	**30.19**	**29.93**	**29.64**	–

RCNN demosaicking [54] (*Park+RCNN*).

– The PCA-CFA filter proposed in [58] is a spatially-adaptive denoising based on principal component analysis (PCA) that exploits the spatial and spectral correlations of CFA images to preserve color edges and details. We evaluate PCA-CFA [58] followed by DLMM demosaiciking [59] (*PCA+DLMM*) and RCNN demosaicking [54] (*PCA+RCNN*).

– Finally, as a reference, we include the deep learning based joint denoising and demosaicking (*JCNN*) of [19, 20]. But its results are only available for noise with $\sigma \leq 20$ because the network is not trained beyond that level.

From Tables 6 and 7 we see that the $DM\&DN$ method RCNN+CBM3D as well as RCNN+nlBayes yield the best results on the Kodak dataset, and the margin with respect to the best $DN\&DM$ method (BM3D+RCNN, i.e. BM3D-CFA [13] with RCNN [54]) is quite large: more than 1.5dB on average. In Fig. 3 we compare some results obtained on an image from the Kodak dataset. From the upper-left extract we can see that textures are better restored with RCNN+CBM3D and MLRI+CBM3D, while JCNN introduces some defects. From the extract we see that the $DM\&1.5DN$ methods preserve much more details than BM3D+RCNN, and the result is comparable to JCNN.

On the Imax database RCNN+CBM3D has the highest CPSNRs on high noise levels, by a small gap though. For low noise levels BM3D+RCNN is better, but the difference with RCNN+CBM3D is very small. The joint denoising-demosaicking network JCNN [20] yield the best results on the Imax dataset for $\sigma \leq 20$ (not trained above those levels) yet, the margin with respect to RCNN+CBM3D is again small. Overall, by looking at the average CPSNR we can say that the $DM\&1.5DN$ scheme RCNN+CBM3D is indeed much more robust than BM3D+RCNN.

Evaluation on real images. We evaluated on a set of 14 raw images taken from the Small SIDD dataset [1]. For simplicity, the selected images correspond to phones from the same manufacturer. We adopted the simple pipeline proposed by the authors, which yields photo finished images that can be compared with the ground truth. The considered methods (RCNN+CBM3D, CBM3D+RCNN, and JCNN) were applied at the demosaicking stage (in linear space). Before any denoising step (DN) we applied a VST (squared root [5]), which whitens the noise, and invert it afterwards. The noise level was estimated using [11] and provided to the denoising algorithms and JCNN.

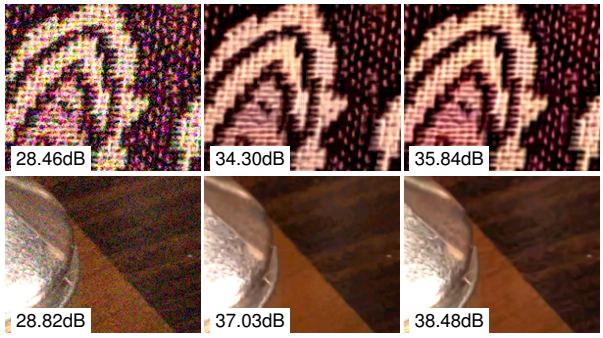


Figure 5: Details of a real images (enhanced contrast) from the SIDD [1] dataset. From left to right: noisy input (demosaicked), BM3D+RCNN , and RCNN+CBM3D.

Table 8: Average CPSNR over 14 raw images taken from the Small SIDD dataset [1]. The reported average noise level is scaled to the range 0-255.

mean σ	CBM3D+RCNN	RCNN+CBM3D	JCNN
7.65	38.19	**39.64**	38.54

Table 8 reports the average CPSNR obtained on these images and the average of the estimated noise levels (after whitening). These values are consistent with the simulated results obtained on the Kodak database (Table 7). The result in Fig. 5, and the supplementary material, support the case in favor of the $DM\&1.5DN$ schemes (RCNN+CBM3D).

4 Conclusions

This paper analyzed the advantages and disadvantages of denoising before demosaicking ($DN\&DM$) schemes, versus demosaicking before denoising ($DM\&DN$), to recover high quality full-color images. We showed that for the $DM\&DN$ schemes a very simple change of the noise parameter of the denoiser DN coped with the structure of demosaicked noise, and led to efficient denoising *after* demosaicking. We found that, this allowed to preserve fine structures that are often smoothed out by the $DN\&DM$ schemes. Our best performing combination in terms of quality and speed is a $DM\&1.5DN$ scheme, where demosaicking DM is done by a fast algorithm RCNN [54] followed by CBM3D denoising $1.5DN$ with noise parameter equal to $1.5\sigma_0$.

Nevertheless it seems ineluctable to see deep learning win the end game when solutions will be found to have more compact or more rapid joint demoisaicking-denoising algorithms.

Acknowledgments : Work partly financed by Office of Naval research grant N00014-17-1-2552 and DGA Astrid project n° ANR-17-ASTR-0013-01.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S. Brown. A High-Quality Denoising Dataset for Smartphone Cameras. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1692–1700. IEEE, jun 2018.

[2] Jan Aelterman, Bart Goossens, Jonas De Vylder, Aleksandra Pižurica, and Wilfried Philips. Computationally efficient locally adaptive demosaicing of color filter array images using the dual-tree complex wavelet packet transform. *PloS one*, 8(5):e61846, 2013.

[3] Hiroki Akiyama, Masayuki Tanaka, and Masatoshi Okutomi. Pseudo four-channel image denoising for noisy cfa raw data. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4778–4782. IEEE, 2015.

[4] David Alleysson, Sabine Susstrunk, and Jeanny Hérault. Linear demosaicing inspired by the human visual system. *IEEE Transactions on Image Processing*, 14(4):439–449, 2005.

[5] F. J. Anscombe. The Transformation of Poisson, Binomial and Negative-Binomial Data. *Biometrika*, 35(3/4):246, dec 1948.

[6] Bryce E Bayer. Color imaging array, July 20 1976. US Patent 3,971,065.

[7] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

[8] Antoni Buades, Bartomeu Coll, Jean-Michel Morel, and Catalina Sbert. Self-similarity driven demosaicking. *Image Processing On Line*, 1:51–56, 2011.

[9] Priyam Chatterjee, Neel Joshi, Sing Bing Kang, and Yasuyuki Matsushita. Noise suppression in low-light images through joint denoising and demosaicing. In *CVPR 2011*, pages 321–328. IEEE, 2011.

[10] Xiangdong Chen, Liwen He, Gwanggil Jeon, and Jechang Jeong. Multidirectional weighted interpolation and refinement method for bayer pattern cfa demosaicking. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(8):1271–1282, 2015.

[11] Miguel Colom and Antoni Buades. Analysis and Extension of the Ponomarenko et al. Method, Estimating a Noise Curve from a Single Image. *Image Processing On Line*, 3:173–197, 2013.

[12] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising with block-matching and 3d filtering. In *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, page 606414. International Society for Optics and Photonics, 2006.

[13] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Color image denoising via sparse 3d collaborative filtering with grouping constraint in luminance-chrominance space. In *2007 IEEE International Conference on Image Processing*, volume 1, pages I–313. IEEE, 2007.

[14] Aram Danielyan, Markku Vehvilainen, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Cross-color bm3d filtering of noisy raw data. In *2009 international workshop on local and non-local approximation in image processing*, pages 125–129. IEEE, 2009.

[15] Weishong Dong, Ming Yuan, Xin Li, and Guangming Shi. Joint demosaicing and denoising with perceptual optimization on a generative adversarial network. *arXiv preprint arXiv:1802.04723*, 2018.

[16] Eric Dubois. Frequency-domain methods for demosaicking of bayer-sampled color images. *IEEE Signal Processing Letters*, 12(12):847–850, 2005.

[17] Thibaud Ehret, Axel Davy, Pablo Arias, and Gabriele Facciolo. Joint demosaicing and denoising by overfitting of bursts of raw images. In *ICCV 2019*, 2019.

[18] Thibaud Ehret, Axel Davy, Jean-Michel Morel, Gabriele Facciolo, and Pablo Arias. Model-blind Video Denoising Via Frame-to-frame Training. In *CVPR 2019*, pages 11369–11378, 2019.

[19] Thibaud Ehret and Gabriele Facciolo. A Study of Two CNN Demosaicking Algorithms. *Image Processing On Line*, 9:220–230, 2019.

[20] Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (TOG)*, 35(6):191, 2016.

[21] Bart Goossens, Hiep Luong, Jan Aelterman, Aleksandra Pizurica, and Wilfried Philips. An overview of state-of-the-art denoising and demosaicking techniques: toward a unified framework for handling artifacts during image reconstruction. In *Image Sensor Workshop*, 2015.

[22] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869, 2014.

[23] John F Hamilton Jr and James E Adams Jr. Adaptive color plan interpolation in single sensor color electronic camera, May 13 1997. US Patent 5,629,734.

[24] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *IEEE transactions on pattern analysis & machine intelligence*, (6):1397–1409, 2013.

[25] Felix Heide, Markus Steinberger, Yun Ta Tsai, Mushfiqur Rouf, and Kari Pulli. Flexisp: A flexible camera image processing framework. *Acm Transactions on Graphics*, 33(6):231:1–231:13, 2014.

[26] Keigo Hirakawa and Thomas W Parks. Joint demosaicing

and denoising. *IEEE Transactions on Image Processing*, 15(8):2146–2157, 2006.

[27] Viren Jain and Sebastian Seung. Natural image denoising with convolutional networks. In *Advances in neural information processing systems*, pages 769–776, 2009.

[28] Sunil Prasad Jaiswal, Oscar C Au, Vinit Jakhetiya, Yuan Yuan, and Haiyan Yang. Exploitation of inter-color correlation for color image demosaicking. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 1812–1816. IEEE, 2014.

[29] Qiyu Jin, Ion Grama, Charles Kervrann, and Quansheng Liu. Nonlocal means and optimal weights for noise removal. *SIAM Journal on Imaging Sciences*, 10(4):1878–1920, 2017.

[30] Qiyu Jin, Ion Grama, and Quansheng Liu. Convergence theorems for the non-local means filter. *Inverse Problems & Imaging*, 12(4):853–881, 2018.

[31] Ossi Kalevo and Henry Rantanen. Noise reduction techniques for bayer-matrix images. In *Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications III*, volume 4669, pages 348–359. International Society for Optics and Photonics, 2002.

[32] Daniel Khashabi, Sebastian Nowozin, Jeremy Jancsary, and Andrew W Fitzgibbon. Joint demosaicing and denoising via learned nonparametric random fields. *IEEE Transactions on Image Processing*, 23(12):4968–4981, 2014.

[33] Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Residual interpolation for color image demosaicking. In *2013 IEEE International Conference on Image Processing*, pages 2304–2308. IEEE, 2013.

[34] Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Minimized-laplacian residual interpolation for color image demosaicking. In *Digital Photography X*, volume 9023, page 90230L. International Society for Optics and Photonics, 2014.

[35] Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Beyond color difference: Residual interpolation for color image demosaicking. *IEEE Transactions on Image Processing*, 25(3):1288–1300, 2016.

[36] Yonghoon Kim and Jechang Jeong. Four-direction residual interpolation for demosaicking. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(5):881–890, 2016.

[37] Teresa Klatzer, Kerstin Hammernik, Patrick Knobelreiter, and Thomas Pock. Learning joint demosaicing and denoising based on sequential energy minimization. In *2016 IEEE International Conference on Computational Photography (ICCP)*, pages 1–11. IEEE, 2016.

[38] Filippos Kokkinos and Stamatios Lefkimmiatis. Iterative joint image demosaicking and denoising using a residual denoising network. *IEEE Transactions on Image Processing*, 2019.

[39] Marc Lebrun, Antoni Buades, and Jean-Michel Morel. A nonlocal bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013.

[40] Min Lee, Sang Park, and Moon Kang. Denoising algorithm for cfa image sensors considering inter-channel correlation. *Sensors*, 17(6):1236, 2017.

[41] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: Learning Image Restoration without Clean Data. In *35th International Conference on Machine Learning, ICML 2018*, 2018.

[42] Julien Mairal, Francis R Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *ICCV*, volume 29, pages 54–62. Citeseer, 2009.

[43] Henrique S Malvar, Li-wei He, and Ross Cutler. High-quality linear interpolation for demosaicing of bayer-patterned color images. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages iii–485. IEEE, 2004.

[44] Daniele Menon and Giancarlo Calvagno. Joint demosaicking and denoisingwith space-varying filters. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 477–480. IEEE, 2009.

[45] Yusuke Monno, Daisuke Kiku, Masayuki Tanaka, and Masatoshi Okutomi. Adaptive residual interpolation for color and multispectral image demosaicking. *Sensors*, 17(12):2787, 2017.

[46] Dmitriy Paliy, Mejdi Trimeche, Vladimir Katkovnik, and Sakari Alenius. Demosaicing of noisy data: spatially adaptive approach. In *Image Processing: Algorithms and Systems V*, volume 6497, page 64970K. International Society for Optics and Photonics, 2007.

[47] Sung Hee Park, Hyung Suk Kim, Steven Lansel, Manu Parmar, and Brian A Wandell. A case for denoising before demosaicking color filter array data. In *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, pages 860–864. IEEE, 2009.

[48] Sukanya Patil and Ajit Rajwade. Poisson noise removal for image demosaicing. In *BMVC*, 2016.

[49] Ibrahim Pekkucuksen and Yucel Altunbasak. Gradient based threshold free color filter array interpolation. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 137–140. IEEE, 2010.

[50] Nikolay Ponomarenko, Vladimir V. Lukin, Mikhail Zriakhov, Arto Kaarna, and Jaakko T. Astola. An automatic approach to lossy compression of AVIRIS images. In *Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International*, pages 472–475. IEEE, 2007.

[51] Javier Portilla, Vasily Strela, Martin J Wainwright, and

Eero P Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Trans Image Processing*, 12(11), 2003.

[52] Nai-Sheng Syu, Yu-Sheng Chen, and Yung-Yu Chuang. Learning deep convolutional networks for demosaicing. *arXiv preprint arXiv:1802.03769*, 2018.

[53] Daniel Stanley Tan, Wei-Yang Chen, and Kai-Lung Hua. Deepdemosaicking: Adaptive image demosaicking via multiple deep fully convolutional networks. *IEEE Transactions on Image Processing*, 27(5):2408–2419, 2018.

[54] Runjie Tan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Color image demosaicking via deep residual learning. In *IEEE Int. Conf. Multimedia and Expo (ICME)*, 2017.

[55] Lei Wang and Gwanggil Jeon. Bayer pattern cfa demosaicking based on multi-directional weighted interpolation and guided filter. *IEEE Signal Processing Letters*, 22(11):2083–2087, 2015.

[56] Jiqing Wu, Radu Timofte, and Luc Van Gool. Demosaicing based on directional difference regression and efficient regression priors. *IEEE transactions on image processing*, 25(8):3862–3874, 2016.

[57] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

[58] Lei Zhang, Rastislav Lukac, Xiaolin Wu, and David Zhang. Pca-based spatially adaptive denoising of cfa images for single-sensor digital cameras. *IEEE transactions on image processing*, 18(4):797–812, 2009.

[59] Lei Zhang and Xiaolin Wu. Color demosaicking via directional linear minimum mean square-error estimation. *IEEE Transactions on Image Processing*, 14(12):2167–2178, 2005.

[60] Lei Zhang, Xiaolin Wu, Antoni Buades, and Xin Li. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic imaging*, 20(2):023016, 2011.

[61] Xingyu Zhang, Ming-Ting Sun, Lu Fang, and Oscar C Au. Joint denoising and demosaicking of noisy cfa images based on inter-color correlation. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5784–5788. IEEE, 2014.

View publication stats